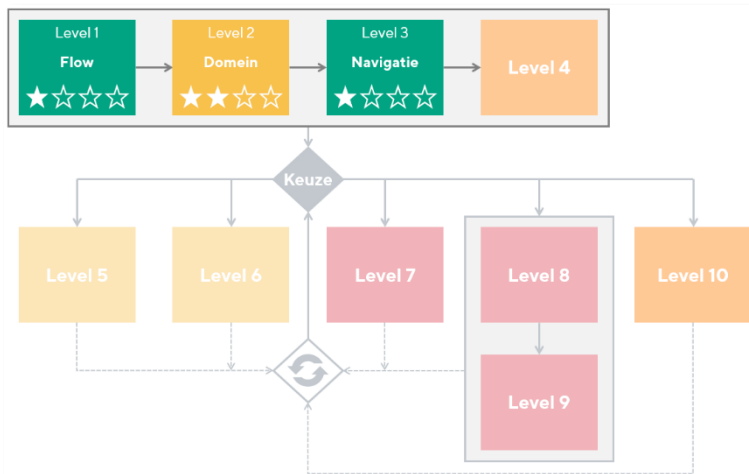


LEVEL 3: NAVIGEREN TUSSEN TWEE PAGINA'S

NIVEAU: ★☆☆☆

Dit derde level heeft als eindresultaat dat we tussen twee pagina's kunnen navigeren. Daarvoor hebben we buttons, events en een tweede pagina nodig.



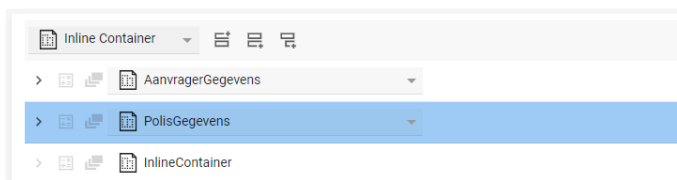
Onze huidige flow heeft slechts één pagina. In dit level passen we die flow aan, zoals weergegeven aan de linkerkant. De flow en de pagina met de verzekeringsaanvraag is al gemaakt, dus dat hoeft niet nog een keer.

Op de eerste pagina kan de klant zijn of haar persoonlijke, levensstijl- en dekkingsdetails invullen. Op deze pagina wordt ook de premie van de polis weergegeven.

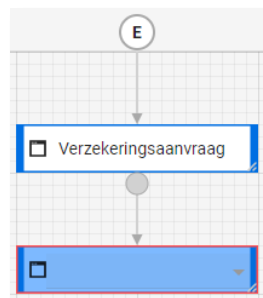
Daarna zullen we een pagina weergeven met alleen bedankbericht. De klant moet heen en weer kunnen navigeren tussen pagina's, zoals de pijlen aangeven.

1. Maak twee **events** (die vind je in het menu **Flow**), een event om verder te gaan en eentje om terug te gaan. Bij een event is verder niks nodig, behalve een naam. Het veld **Parameters** kun je leeglaten. Maak ook twee **buttons** (die vind je in het menu **Interaction**): Een button om verder te gaan en een om terug te gaan. Vergeet de **Caption** niet want die zie je straks in de applicatie.

Open je eerste **pagina** en voeg de **Verder** button toe in een inline container, zet die onderaan. Dit doe je door op **PolisGegevens** te gaan staan, op het middelste icoon (insert after) te klikken nadat je gekozen hebt voor **Inline Container**. Bij een button kun je verschillende opties ingeven. Zo moet je het bijbehorende **event** kiezen en het **event type**. Het event ligt voor de hand, laat het event type staan op **Validate and continue**.



2. Bewaar de pagina en open de **flow**. Er is nu een stip verschenen aan de pagina. Deze stip komt overeen met het event dat je zojuist gemaakt hebt (en indirect dus de button, want daar hangt dit event aan). Deze stip kun je gebruiken om straks een pijl naar je tweede pagina te modelleren.
3. Maak eerst een tweede **pagina** aan en zet Dankjewel (of iets dergelijks) als **Display name**.





De pagina heeft een **button** nodig om weer terug te keren naar de eerste pagina. Kies voor het **Event type** voor deze **Terug**-button ook weer **Validate and continue**.

Pas de **flow** aan zodat je kan navigeren tussen je twee pagina's. Je moet dus van de pagina **Verzekeringsaanvraag** naar **Dankjewel** kunnen navigeren en weer terug.

Test je applicatie weer in de Runtime.
Kun je mooi heen en weer tussen je pagina's?

4. De naam van de aanvrager is verplicht. Veel mensen denken dat je dit moet doen bij de definitie van het **attribuut**, maar dan zou dit attribuut op alle pagina's waar het gebruikt wordt verplicht zijn. Dat is niet wenselijk, dus moet je deze eigenschap aanpassen in de **container** waar het attribuut voorkomt. Pas dit aan (verplicht is **Required**) en bekijk het resultaat in de Runtime.
5. Maak de currency attributen **Polis.PremieBasisVerzekering**, **Polis.PremieAanvullendeVerzekering**, **Polis.Korting** en **Polis.PremiePerMaand** **Read only**.

Dat moet je om dezelfde reden als bij verplichtheid dus ook in de container doen en niet bij het attribuut zelf. Bekijk het resultaat weer in de runtime.

6. Valt het je op dat de niet ingevulde **Read only attributen** lijken te verdwijnen? Dat komt omdat ze helemaal leeg zijn. Je kunt dit aanpassen door ze een **default value** te geven, bijvoorbeeld 0 euro voor de premie van de aanvullende verzekeringen.

Attribute		Default value
Entity Polis	Name PremieAanvullendeVerzekering	Default value None
Data type Currency	Value list	
☐ Multivalued		